In the Claims

    Claims pending

- At time of the Action: Claims 1-28.

- After this Response: Claims 1, 2, 4-12, 14-17, and 19-28.

    Canceled claims: Claims 3, 13, and 18.

    Amended claims: Claims 1, 4, 5, 6, 8, 12, 17, 19, 20, 21, 23, and 25.

    New claims: None.

1.   (Currently Amended) A method of generating common intermediate language code comprising:

    writing first ~~JAVA~~™ <u>object oriented</u> language source code that comprises a definition of a generic class usable in a framework;

    generating an instance of the generic class; ~~and~~

    compiling the instance of the generic class into common intermediate language code executable by a runtime engine; <u>and</u>

    <u>receiving second object oriented language source code referencing the generic class defined by the first object oriented language source code.</u>

2.   (Original) A method as recited in claim 1 further comprising storing the source code in a class library of the framework.

3.   (Canceled).

4.   (Currently Amended) A method as recited in claim 1 further comprising:

~~receiving second source code referencing the generic class; and~~

parsing the second source code into a parse tree representing the second source code.

5.   (Currently Amended) A method as recited in claim 1 further comprising parsing the portion of ~~JAVA™~~ the object oriented language source code into a parse tree representing the source code.

6.   (Currently Amended) A method as recited in claim 1 wherein writing first ~~JAVA™~~ object oriented language source code comprises defining at least one parameter associated with the generic class.

7.   (Original)   A method as recited in claim 6 wherein the at least one parameter is an unconstrained type.

8.   (Currently Amended) A method as recited in claim 1 further comprising declaring an instance of the generic class in second ~~JAVA™~~ object oriented language source code.

9.   (Original) A method as recited in claim 8 wherein declaring an instance of the generic class comprises specifying a type from a plurality of allowable types associated with the generic class.

10.   (Original) A method as recited in claim 9 wherein the specified type is another generic class.

11.   (Original) A method as recited in claim 1 wherein the generic class comprises one of:

a Queue class;

a Dictionary class; and

a Stack class.

12.   (Currently Amended) A method of using a generic class comprising:

adapting existing ~~JAVA™~~ object oriented language source code to include a declaration of a first generic class provided by a software package having a class definition of the first generic class wherein the adapting comprises editing the existing object oriented language source code with a second source in a second source framework; and

compiling the adapted ~~JAVA™~~ object oriented language source code with the class definition to generate common intermediate language code.

13. (Canceled).

14. (Original) A method as recited in claim 12 wherein the class definition defines at least one parameter of the generic class.

15. (Original) A method as recited in claim 12 wherein compiling comprises:

validating a specified type of the generic class according to the class definition.

16. (Original) A method as recited in claim 12 wherein the adapting comprises nesting a second generic class in the declaration of the first generic class.

17. (Currently Amended) A system for authoring source code comprising:

a class library having a generic class definition; ~~and~~

a means for receiving a declaration of an instance of the generic class in ~~JAVA™~~ first object oriented language source code wherein the means for receiving comprises a computer-readable medium having stored thereon a second source application; and

a means for generating metadata descriptive of the generic class.

18.   (Canceled).

19.   (Currently Amended) A system as recited in claim 17 further comprising a common intermediate language importer operable to associate the generic class declaration in the ~~JAVA™~~ object oriented language source code to the generic class definition.

20.   (Currently Amended) A system as recited in claim 17 further comprising a semantic analyzer operable to validate the generic class declaration in the ~~JAVA™~~ object oriented language source code according to the generic class definition.

21.   (Currently amended) A system as recited in claim 17 further comprising a code generator ~~operable to generate metadata descriptive of the generic class and further~~ operable to generate common intermediate language code representative of the generic class.

22.   (Original) A system as recited in claim 21 further comprising a runtime engine operable to translate the common intermediate language into machine-specific binary executable by a computer associated with the runtime engine.

23.   (Currently Amended) A computer-readable medium having stored thereon microprocessor-executable instructions for performing a method comprising:

receiving input representing a generic class definition in a ~~JAVA™~~ object oriented language;

receiving source code that references the generic class; ~~and~~

compiling the source code with an instance of the generic class into common intermediate language code executable by a runtime engine; <u>and</u>

<u>receiving second object oriented language source code referencing the generic class defined by the first object oriented language source code.</u>

24. (Original) A computer-readable medium as recited in claim 23 wherein the method further comprises storing the generic class definition in a framework class library.

25. (Currently Amended) A computer-readable medium as recited in claim 23 wherein the source code comprises ~~JAVA™~~ <u>object oriented</u> language source code.

26. (Original) A computer-readable medium as recited in claim 23 wherein the method further comprises generating metadata describing the generic class.

27. (Original) A computer-readable medium as recited in claim 23 wherein the generic class definition comprises a generic class name and two angular brackets around one or more parametric types.

28. (Original) A computer-readable medium as recited in claim 23 wherein the method further comprises compiling the generic class definition into common intermediate language code.